

APPLE-AIDS
APPLE-AIDS
APPLE-AIDS
APPLE-AIDS
APPLE-AIDS
APPLE-AIDS
APPLE-AIDS

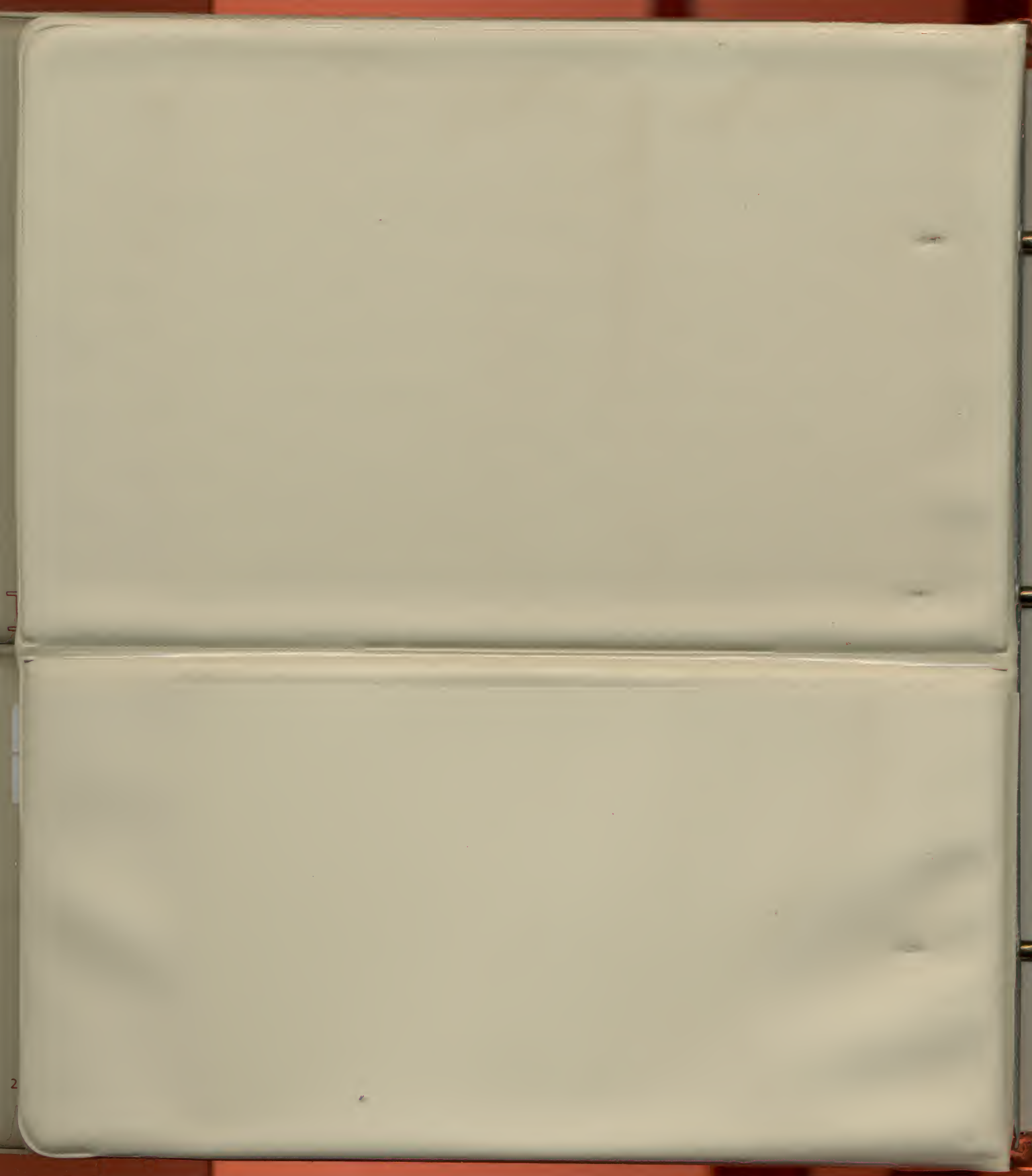
A collection of 12
extremely useful
utilities
by ALLEN WYATT

Kill DOS • Disk Map • Undelete Files • Disk Editor • Text Dump • Base Convert



HOWARD W. SAMs

Developed by Advanced Operating Systems



LIMITED WARRANTY

For a period of ninety (90) days from the date of original purchase at retail, the warrantor, identified below, warrants this software to load and run as a basic program for the indicated microcomputer model, to be free from defects in material and workmanship and to be merchantable and suitable for its stated purpose for the period of this warranty. This warranty may not be enlarged except in writing, signed by warrantor. A Customer Registration Card is enclosed in this package. Please complete and return the Card to the warrantor within thirty (30) days. THE WARRANTOR EXPRESSLY DISCLAIMS ANY IMPLIED WARRANTY INCLUDING THE WARRANTY OF MERCHANTABILITY AND THE WARRANTY THAT THE SOFTWARE IS SUITABLE FOR ITS STATED PURPOSE AS OF THE DATE NINETY (90) DAYS FROM THE ORIGINAL PURCHASE OF THE SOFTWARE AT RETAIL.

•

In the event of defect, malfunction or failure of the software to conform with this warranty, the warrantor will repair or replace the software at no cost to you. For warranty service, you should return the software to the warrantor, Howard W. Sams & Co., Inc., 4300 W. 62nd Street, Indianapolis, Indiana 46268. Software received damaged as a result of shipping will require you to file a claim with the carrier. This warranty gives you specific legal rights and you may also have some other rights which vary from state to state.

•

THIS WARRANTY IS LIMITED SOLELY TO THE ABOVE AND THIS WARRANTY AND ANY WARRANTIES IMPLIED BY STATE LAW WILL APPLY ONLY FOR THE PERIOD SET FORTH. (SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.) THE WARRANTOR WILL NOT BE LIABLE FOR ANY LOSS, DAMAGE, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND, WHETHER BASED UPON WARRANTY, CONTRACT OR NEGLIGENCE, AND ARISING IN CONNECTION WITH THE SALE, USE OR REPAIR OF THE SOFTWARE. (SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.) UNLESS OTHERWISE CONTRARY TO STATE LAW GOVERNING THE PURCHASE, THE WARRANTOR'S LIABILITY SHALL NOT IN ANY CASE EXCEED THE CONTRACT PRICE FOR THE SOFTWARE CLAIMED TO BE DEFECTIVE OR UNSUITABLE.

•

WARNING: THE UNAUTHORIZED USE, REPRODUCTION OR DUPLICATION OF THIS MATERIAL, OR ITS PUBLIC PERFORMANCE OR DISPLAY, BY ANY MEANS IN ANY MEDIA FOR ANY PURPOSE, WHETHER IN WHOLE OR IN PART, IS STRICTLY PROHIBITED. VIOLATORS WILL BE SUBJECT TO ALL CIVIL AND CRIMINAL PENALTIES.

HOWARD W. SAMS & CO., INC.

4300 W. 62nd Street
Indianapolis, Indiana 46268

APPLE-AIDS

by Allen L. Wyatt

LOADING INSTRUCTIONS

Getting Started

In order to use APPLE-AIDS, you must have a 48K Apple II® or Apple II PLUS® computer. You must also have at least one disk drive.

APPLE-AIDS is written in Applesoft® and Machine Language. In order to use APPLE-AIDS properly, you must boot your system up using your APPLE-AIDS disk. You should execute the following steps:

1. Insert your APPLE-AIDS disk into drive 1.
2. Turn your computer power on, or type PR#s ("s" is the slot containing the controller card of the drive from which you wish to boot) and press **RETURN**.
3. You will see the APPLE-AIDS Main Program Menu appear on your CRT.

APPLE AIDS / PROGRAMMING HELPS
BY ALLEN L. WYATT
(C) 1981 BY ADVANCED OPERATING SYSTEMS

1) FORMAT BLANK DISK
2) KILL DOS
3) DISK MAP
4) IMPROVED DIRECTORY
5) SECTOR LISTINGS
6) UNDELETE FILES
7) DISK EDITOR
8) DISK COPY
9) CREATE EXEC FILES
10) EDIT EXEC FILES
11) TEXT DUMP
12) BASE CONVERT
13) EXIT SYSTEM

>CHOICE: (RETURN TO RUN)

You are now ready to use any of the programs you see listed on your CRT. If the preceding three steps fail, check all of your electrical connections (including pc board contacts) and repeat the steps again.

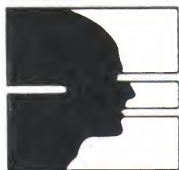
After you see the Main Program Menu, it would be a good idea to make a back-up copy of your APPLE-AIDS program disk. You can do this by choosing option 8 from the menu. Chapter 9 in this manual explains making back-up copies in more detail.

APPLE-AIDS will work with any Apple II® computer that has Applesoft in ROM® or a Language Card. If you have a standard Apple II®, with a Language Card, please do the following:

1. Insert a System Master diskette into your drive.
2. Boot up to this disk so that Applesoft® will be loaded into your Language Card.
3. Insert your APPLE-AIDS disk, type PR#6 and press **RETURN** .

If all else fails, contact your dealer. They can help you with most any problem you may encounter.

TABLE OF CONTENTS



APPLE-AIDS

• Introduction	7
• Format Blank Disk	9
• Kill DOS	11
• Disk Map	13
• Improved Directory	15
• Sector Listings	17
• Undelete Files	19
• Disk Editor	21
• Disk Copy	31
• Create Exec Files	33
• Edit Exec Files	35
• Text Dump	37
• Base Convert	39
• Exit System	41
• User Notes	43
• Glossary	44
• Appendix A—VTOC	47
• Appendix B—Diskette Directory	49
• Appendix C—Track Sector Lists	52
• Appendix D—How DOS Deletes A File	54
• Appendix E—Disk Potpourri	56
• Appendix F—Apple ASCII Chart	57
• Appendix G—ACSII Code Table	58
• Index	61



Chapter 1

INTRODUCTION

It has always been important for a personal computer user to be able to have access to “real” information. That information might be timely data, or it could be hard-to-find reference material. If there is a way to get such information cheaper or easier, the method of such acquisition generally is sought after.

Easy information retrieval and manipulation was the reason for this software package. At the time, there was no software package on the market that allowed the user to do all the things that needed to be done in software development. Thus, a set of tools was developed and refined to help in the development of other software products. These tools needed to be powerful, fast, friendly, and reliable.

This is the idea behind Apple-Aids. We think you will find them easy to use and helpful to you in your understanding and development of software. Before you start, however, we need to give a word of caution.



Please be extremely careful in your use of these utilities. Just as with all powerful tools, there is the ability to destroy as well as create. **ALWAYS MAKE BACK-UP COPIES OF ALL WORK!** This way, if something does go wrong, you are not left with a lot of ruined work. Make sure you understand **EXACTLY** what you intend to do, as well as what should happen, before you attempt to do it.

This manual has been written, for the most part, with the idea that the user would be using it on a DOS 3.3 system. If you are using it on 3.2, all operations will be the same with the exception that you will not be able to access anything above sector 12 (\$C) because there are only 13 sectors on a DOS 3.2 initialized diskette.

As a final note, please read the Loading Instructions “GETTING STARTED” before attempting to use Apple-Aids. It may save you some time.



Chapter 2

FORMAT BLANK DISK

When you format a disk using the "INIT" command of DOS, your Apple writes a series of identifying marks on the diskette so that it can find its way around when it needs to. After doing this, the disk is left completely blank (for a short time). At this point, there are 35 tracks that have been written, each consisting of 16 sectors (or 13 sectors for DOS 3.2). This totals (by simple math), to 560 (or 455) blank sectors. Then "INIT" continues and marks off 64 (or 52) sectors for internal usage. Tracks 0, 1, and 2 are used for an image of the Disk Operating System, and track 17 (\$11) is used for the VTOC and the CATALOG. Then the "HELLO" program you have specified is linked and saved so that it runs when you boot DOS. However, all this leaves you 64 (or 52) sectors short of full disk usage.

This program will format your diskette in a slightly different manner. What happens is that the disk is marked into tracks and sectors, and no image of DOS is saved on the disk. However, track 0 and 17 (\$11) are still "off limits" to a user. Track 0 is not accessible (except when booting or seeking directly, as with the Disk Editor), and track 17 (\$11) is still required so that DOS can keep track of what you have saved on the disk.

To run the program, choose option 1 from the Main Program Menu. Then decide which Slot and Drive from which you wish to format, and follow the instructions.

An important thing to remember is that you will not be able to boot your system using your newly created data disk, since there is no DOS image on the disk. If you do attempt to boot to it, a message will be displayed on the screen notifying you of your error. In order to recover from this point, you will need to insert another system disk and type PR#6 to reboot the system again. Other than that, the disk will behave and appear as normal, except that the storage capacity has been increased by two tracks.

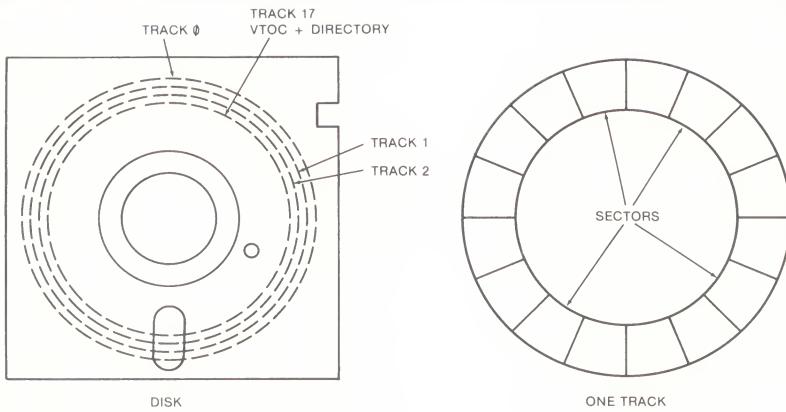


Fig. 1—Track and Sector locations on a disk.



Chapter 3

KILL DOS

KILL DOS will allow you to accomplish the same objective as FORMAT BLANK DISK, but in a slightly different manner. This program assumes that you already have a normally formatted diskette and that you have saved things to the disk already. In fact, you may have saved so much on the disk that you have found that you have run out of room to store some data you need.

Never fear! KILL DOS will allow you to take DOS off of your diskette without harming so much as a single byte of your valuable information. Again, this will “free-up” 32 sectors for your use, but you will not be able to boot your system using your disk (on the Apple, it is a pretty good trick to be able to boot without something with which to boot).

Once you have removed DOS, there are only two ways you can restore it. One is to re-initialize the diskette. The problem with this, though, is that you will lose all of the data you have on your diskette. You could also use “Master Create” (supplied by Apple), but if you have saved any information on tracks 1 or 2, you will still lose that data. So, make sure you want to KILL DOS before you actually do it.

To run this function, simply choose option 2 from the Main Program Menu, and determine on which Slot and Drive you wish to perform the operation. If you have more than one drive on your system, take care that you are “killing” the DOS on the correct drive.

As it is with FORMAT BLANK DISK, you will not be able to boot to your data disk any more. If you attempt to do so, you will be given an error message and returned to BASIC. All you need to do then is reboot to another disk with DOS.



Chapter 4

DISK MAP

Many times, especially when doing highly specialized application software, you need to see exactly which sectors are in use or free on a given diskette. DISK MAP allows you to determine this information from any normally formatted diskette.

To use this utility, choose option 3 from the Main Program Menu. Then simply determine the Slot and Drive you wish to map and press **RETURN** when so prompted. Be patient, it takes a few seconds to map the disk completely.


Now you will see a graphic representation of the space allocation on the disk. "In use" sectors will be denoted with an asterisk (*). On this graph, tracks are represented on the horizontal axis, and sectors on the vertical axis. Hence, if you wish to determine if track





13 (\$D), sector 15 (\$F) is in use, you would locate the track across the top of the chart, and the sector along the side. If there is an asterisk (*) where the row and column meet, then the sector is in use.

DISK MAP also keeps count for you, and will let you know total sectors free as well as total sectors in use. This is handy to determine if you have enough room on a disk to transfer a file completely to the disk. These totals can be found at the bottom of the screen.

At the end of the mapping session, you have the option to map another disk. If you answer by pressing  , you will be returned to the Main Program Menu.



Chapter 5

IMPROVED DIRECTORY

How many times have you wished you knew exactly what was on the catalog of a disk? IMPROVED DIRECTORY will let you see this information.

To use IMPROVED DIRECTORY, choose option 4 from the Main Program Menu. As usual, you will be given a choice as to the Slot and Drive you wish to utilize. When you have entered the appropriate responses, simply press **RETURN** and you will be shown the file name (truncated at 10 characters), the file type (A=Applesoft, I=Integer, B=Binary, T=Text, S=S type, R=Relocatable file, ?=Who Knows), and the file code.

The file code is a one character notation of the status of the file. A file may either be unlocked, locked, or deleted. A blank, or no

```
DISK DIRECTORY - TRK: 17  SECT: 14
CODES:      *=LOCKED      $ = DELETED

NAME: DATA DISK  TYPE: B  CODE: *
      START: 37888      LEN: 399

NAME: SCREEN      TYPE: B  CODE: *
      START: 37904      LEN: 423

NAME: FORMAT BLA  TYPE: A  CODE: *
NAME: KILL DOS    TYPE: A  CODE: *
NAME: DISK MAP    TYPE: A  CODE: *
NAME: IMPROVED D  TYPE: A  CODE: *
NAME: SECTOR LIS  TYPE: A  CODE: *

PRESS 'RETURN' TO CONTINUE....■
```

mark, denotes that the file is unlocked. An asterisk (*) denotes the file is locked, and a dollar sign (\$) denotes that the file is deleted.

This may appear a bit baffling at first, but DOS does not erase a file name from the directory simply because it has been deleted. For more information on the deletion process, see the reference sections at the end of this manual.

Besides the file code, if the file type is 'B' (Binary), you will be shown the beginning address and file length of the program. Both of these numbers are in decimal notation. This can be very helpful, since this information is not shown with normal "CATALOG" function of DOS.



Chapter 6

SECTOR LISTINGS

Sector listings are very handy if you wish to pin-point (exactly) where a file resides on disk. This program provides you precisely with what it suggests: a list of the tracks and sectors used by the files on a disk.

In order to use this utility, choose option 5 from the Main Program Menu. When you answer the usual questions as to Slot and Drive, and then press **RETURN**, you will be shown a series of numbers in the format (TT-SS) In other words, the track is shown first, and then the sector, so that if you see the numbers (17-15), part of the file you are examining is contained on track 17, sector 15.

The numbers shown in inverse (black on white) denotes that the particular sector shown is used for the track-sector list of the file being examined. All other sectors are shown in normal mode (white on black.)

Also, it is important to remember that all information shown is represented in decimal notation.



Chapter 7

UNDELETE FILES

I don't know about you, but there are many times when I have deleted files that I didn't want to, but once I hit **RETURN**, it was too late. This usually has happened late at night when I've been up for 43 hours straight and am having trouble seeing the monitor, let alone hitting the right keys.

Be that as it may, I am left with no way to recover a file I just deleted, right?

Wrong! All you need to do is to use UNDELETE FILES to salvage that lost program, and it will "magically" reappear in complete form. If you wish to learn how this is done (in detail), turn to the reference sections at the end of this manual.

In order to use the program, choose option 6 from the Main Program Menu. Again, you are given the choice of Slot and Drive. After entering the proper choices, press **RETURN** to begin the review. As each previously deleted program is located in the directory, you will be shown the file name and given a choice whether to "undelete" the file, or not. If you answer "N", the next file will be searched for. If you answer "Y", the program sets relative byte 0 of the Directory back to its original value, and puts the track bit map back to its "pre-deleted" values. Now the file is resurrected and all is as before.

There are a few pointers to remember, however. If you have deleted a program and later decide to undelete it, this will only be meaningful if you have not saved anything else out to the disk since you last deleted the file. If you did, it is very likely that DOS, thinking you meant it when you deleted the file, has written your new file into part, if not all, of the space previously occupied by the deleted file. If you try to undelete under these circumstances, chances are you will mess up the directory; and that will mean extensive patch work later on, a blown directory, or garbaged files. In any case, it is a good practice to use UNDELETE FILES only immediately after having deleted the file.



Chapter 8

DISK EDITOR

This is the single most used program in our library. DISK EDITOR is a powerful utility that will allow you to load, inspect, alter, and write any sector of any normally formatted disk. To use DISK EDITOR, choose option 7 from the Main Program Menu and you will see the following:



To avoid possible inadvertant damage to your Apple-Aids disk, we suggest that you remove it from the drive at this time. You may then insert any disk that you wish to edit.

All of the DISK EDITOR commands shown in the command line are single-key commands. You do NOT need to press **RETURN**. The commands and their subsequent actions are:

L—This command will **LOAD** any sector you designate into the editor buffer and then display it on the screen in the current display mode (for display modes, see commands A, D, and H).

W—This is the opposite of load, i.e., this command will **WRITE** the sector currently in the editor buffer to any place you designate on the disk.

E—**EDIT**. This will allow you to change any or all of the information displayed on the screen. These changes are made in the editor buffer and can then be written back out to the disk. The editing begins at whatever byte you specify, and you have the choice of several edit types. The edit types are:

D—Edit will be a one-byte **DECIMAL** change.



H—Edit will be a one-byte **HEXADECIMAL** change.

A—Edit will be an **ASCII** string change starting with the specified byte. For instance, with this type of edit you can enter a word as the change, and it will be inserted beginning with the starting byte you specified.


E—Edit will be an **ENHANCED ASCII** string change. This is the same as an **ASCII** change, except that the character values have the high byte on (**ASCII** value + \$80).


F—Edit mode will be **"FILL"**. You will be asked for the ending byte and the type of fill (decimal or hex), and then the value with which to fill. All bytes between the limits you have set will then be changed to the desired value.

All edits are done immediately and displayed in the current display mode.

+—This command will allow you to **INCREMENT** the sector pointer by one, **LOAD** and **DISPLAY** in the current mode. Thus, if you are currently viewing track 3, sector 4 on the display, upon pressing  track 3, sector 5 will be loaded and displayed. Also, track borders are automatically bridged, so that if you are on track 34 (\$22), sector 15 (\$F) and you press , track 0 sector 0 will be loaded and displayed.



—The ‘-’ command works the same as the ‘+’ command, but in reverse. Pressing  will DECREMENT the sector pointer by one, LOAD and DISPLAY in the current mode. Track borders are automatically bridged here, also.

S—This command SWITCHES the part of the sector being viewed on the CRT. Each page of display can accommodate 128 (\$80) bytes of data. The first page display comprises bytes 0-127 (\$0-\$7F). The second contains bytes 128-255 (\$80-\$FF). Pressing  will toggle between the two halves of the sector. The current display mode (A, D, or H) is retained.

A—This displays the current page of the editor buffer in ASCII representation. ASCII characters in the range 0-\$3F and \$80-\$9F are shown in inverse (black on white) print. Characters in the range \$40-\$7F are flashing, and characters between \$A0 and \$FF are shown in normal print. This display mode is maintained until another display mode is chosen.



D—This displays the current page of the editor buffer in DECIMAL notation. This display mode is maintained until another display mode is chosen.

SLOT: 6 DRIVE: 1 TRACK: 17 SECTOR: 14

0	0	17	13	0	0	0	0	0
0	0	0	0	21	15	132	196	193
1	212	193	160	196	201	211	203	160
2	160	160	160	160	160	160	160	160
3	160	160	160	160	160	160	160	160
4	160	160	160	160	160	160	160	160
5	160	160	160	160	3	0	22	15
6	132	211	195	210	197	197	206	160
7	160	160	160	160	160	160	160	160
8	160	160	160	160	160	160	160	160
9	160	160	160	160	160	160	160	160
A	160	160	160	160	160	160	160	3
B	0	23	15	130	198	207	210	205
C	193	212	160	194	204	193	206	203
D	160	196	201	211	203	160	160	160
E	160	160	160	160	160	160	160	160
F	160	160	9	0	24	15	130	203
10	201	204	204	160	196	207	211	160

␣ L,W,E,+,-,S,A,D,H,C,I,P,*,?,X ␣

COMMAND:



H—This displays the current page of the editor buffer in HEX-ADECIMAL notation. This display mode is maintained until another display mode is chosen.

SLOT: 6 DRIVE: 1 TRACK: 11 SECTOR: 0E

00	00	11	00	00	00	00	00	00
00	00	00	00	15	0F	84	C4	C1
10	D4	C1	A0	C4	C9	D3	CB	A0
10	A0	A0	A0	A0	A0	A0	A0	A0
20	A0	A0	A0	A0	A0	A0	A0	A0
30	A0	A0	A0	A0	03	00	16	0F
40	84	D3	C3	D2	C5	C5	CE	A0
50	A0	A0	A0	A0	A0	A0	A0	A0
60	A0	A0	A0	A0	A0	A0	A0	A0
70	A0	A0	A0	A0	A0	A0	A0	03
80	00	17	0F	82	C6	CF	D2	CD
90	C1	D4	A0	C2	CC	C1	CE	CB
A0	A0	C4	C9	D3	CB	A0	A0	A0
B0	A0	A0	A0	A0	A0	A0	A0	A0
C0	A0	A0	09	00	18	0F	82	CB
D0	C9	CC	CC	A0	C4	CF	D3	A0

⌘ L,W,E,+,-,S,A,D,H,C,I,P,*,?,X ⌘

COMMAND: ■

C—This command will CONDENSE the data and show the contents of the entire sector in hex and ASCII format on the CRT at one time. This is for those who may have a need to see the sector displayed as such. This is a viewing mode only. All edits must be done from any of the other display modes (A, D, or H). In order to return to the current display mode, simply press any key.

```

001 001 TRK: 11 SEC: 0E PRESS ANY KEY
002 00110D0000000000000000000015 COMP
003 0F84C4C1D4C1A0C4C9D3CBA0 DATA DISK
004 A0A0A0A0A0A0A0A0A0A0A0A0A0
005 A0A0A0A0A0A0A0A0A00300160F
006 84D3C3D2C5C5CEA0A0A0A0A0
007 A0A0A0A0A0A0A0A0A0A0A0A0
008 A0A0A0A0A0A0A00300170F82
009 C6CFD2CDC1D4A0C2CCC1CECB
010 A0C4C9D3CBA0A0A0A0A0A0A0
011 A0A0A0A0A0A00900180F82CB
012 C9CCCCA0C4CFD3A0A0A0A0A0
013 A0A0A0A0A0A0A0A0A0A0A0A0
014 A0A0A0A0A00700190F82C4C9
015 D3CBA0CDC1D0A0A0A0A0A0A0
016 A0A0A0A0A0A0A0A0A0A0A0A0
017 A0A0A0A009001A0F82C9CDD0
018 D2CFD6C5C4A0C4C9D2C5C3D4
019 CFD2D9A0A0A0A0A0A0A0A0A0
020 A0A0A00B001B0F82D3C5C3D4
021 CFD2A0CCC9D3D4C9CEC7D3A0
022 A0A0A0A0A0A0A0A0A0A0A0A0
023 A0A0A0A0
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053
054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535

```



I—This command will DISSASSEMBLE the current sector buffer. Standard 6502 mnemonics are used and displayed. You can use CTRL-S as a toggle to start/stop the display. To get back to a normal display, simply press any of the display mode commands (A, D, or H).

SLOT: 6 DRIVE: 1 TRACK: 00 SECTOR: 00

9676-	EE	B0	B5	INC	\$B5B0
9679-	D0	03		BNE	\$967E
967B-	EE	BE	B5	INC	\$B5BE
967E-	A9	00		LDA	#\$00
9680-	8D	5D	B6	STA	\$B65D
9683-	4C	46	A5	JMP	\$A546
9686-	8D	BC	B5	STA	\$B5BC
9689-	20	A8	A6	JSR	\$A6A8
968C-	20	EA	A2	JSR	\$A2EA
968F-	4C	7D	A2	JMP	\$A27D
9692-	A0	13		LDY	#\$13
9694-	B1	42		LDA	(\$42),Y
9696-	D0	14		BNE	\$96AC
9698-	C8			INY	
9699-	C0	17		CPY	#\$17
969B-	D0	F7		BNE	\$9694
969D-	A0	19		LDY	#\$19

⌘ L,W,E,+,-,S,A,D,H,C,I,P,*,?,X ⌘

COMMAND: I

P—This is a PRINTER TOGGLE. If you press **P**, you will be asked for your printer slot. All information and output will then be directed to your printer. Pressing **P** again will turn your printer off.

*****—This command will allow you to change the currently logged Slot and Drive on which you are working. A valid slot must be given. Using this command will NOT destroy or clear the buffer.

?—This will show you a LIST OF THE COMMANDS AND EDIT TYPES available. To get back to normal display, simply press any of the display mode commands (A, D, or H).



X—Use this to EXIT the program back to the Main Program Menu.

A word must be said about general editor format. All byte markers are shown to the left side of the screen. These refer to the first byte of the row. There are eight bytes per row, sixteen rows per page, two pages per sector. This is not true in the Condensed viewing mode. In that mode, there are 12 (\$0C) bytes per row, and the entire sector is shown on one page.

Both display and input is determined by the display mode you are in currently. If you are in Hex mode, all display and input is given and expected in Hexadecimal notation. If you are in Decimal or ASCII notation, all display and input are given and expected in decimal notation. The only exception to this is when you are in Edit mode and you are specifying the type of edit to be made. In such cases, you can make a Hexadecimal change or fill while in Decimal or ASCII display modes. You can also make a Decimal or ASCII change or fill while in Hex display mode.

You will notice status markers at the top of the editing screen. These show you on which Slot and Drive you are functioning, and which track and sector is currently in the buffer. If you wish to write a sector back out to the same place from which you loaded it, just enter the same track and sector that you see noted by the indicators.

If you wish to write a sector from one disk to another, simply Load the sector from one disk, then change the Slot and Drive parameters (Use the "*" command). Now write the buffer to any place on the other disk you wish. If you are only using one disk drive, you will need to switch disks physically in your drive to accomplish this same task.

There are no operational differences between the DOS 3.3 and the DOS 3.2 versions of DISK EDITOR. There is one application difference, however, that should be noted.

If you try to read a DOS 3.2 sector that has never had anything written to it, a drive error will be generated. This is due to the fact



that DOS 3.2 leaves a sector COMPLETELY blank until written to. DOS 3.3 does not have this problem, because when DOS 3.3 initializes a disk, it writes zeros to each byte of every sector on the disk, thus no errors are generated when a "blank" 3.3 formatted sector is read. This is only a minor inconvenience of the 3.2 operating system and will in no way affect the other operations of the Editor. You can still write to any of these "unreadable" sectors, and then you will have no problem later reading from them.

Please see the reference sections at the end of this manual for a disk map that shows where certain standard items are located on the disk.



Chapter 9

DISK COPY

This program will copy a disk to another disk. Either one or two disk drives can be used, and there is no need to format the disk being copied onto. This program does that automatically.

All you need to do is choose option 8 from the Main Program Menu. Then determine in which Slot and Drive the original (master) disk will be, and in which Slot and Drive the copy (blank) disk will be. When you press **RETURN**, the disk denoted as the copy (blank) will be formatted and copied to track-by-track.

Notice that you are kept constantly informed regarding the status of the copy procedure. At the successful conclusion of the copy, you will be given the chance to make another copy (parameters can be reset at this time), or to end. If you end, you will be returned to the Main Program Menu.

You can abort a copy session at any time by pressing the **ESC** key. If you do so, the copy process will be halted at the end of the current write cycle and you will be notified that the copy has been aborted. You will also be notified of the last track written to the copy disk.

Let us note here the purpose of copy programs. This copy program is here for your convenience and use. It is intended for you to use in order to make personal back-up copies of your programs. It is NOT meant for you to use to try to copy any copy-righted software for other than personal use. If you do use it for such, you become what is referred to as a "Software Pirate". As such, you are stepping outside the law and your legal rights. If you find it necessary to do so, please be assured that AOS will not support pirated copies of APPLE-AIDS, nor will they support the pirate. If you have any questions about our policy on such matters, please call or write for information.



Chapter 10

CREATE EXEC FILES

Creating Exec files can be a headache. This utility will allow you to put together a series of commands into a text file to be executed later. With a little practice, creating Exec files can become fairly easy to do.

In order to use this utility, choose option 9 from the Main Program Menu. Now you will be asked for the name of the file. Enter any valid file name. It can contain any characters except a comma (,) or colon (:), and it cannot begin with a number. It also cannot be over 29 characters in length.

Next, enter the Slot and Drive on which you will be working. Make sure you have the disk on which you want the Exec file saved in the disk you specify.

CREATE EXEC FILES

THIS PROGRAM WILL ALLOW YOU TO CREATE
EXEC FILES SIMPLY AND ACCURATELY.

YOU WILL BE GIVEN '!' AS A PROMPT. JUST
ENTER WHAT YOU WANT TO BE IN THE FILE
JUST AS YOU WOULD AT BASIC COMMAND
LEVEL.

WHEN YOU ARE THROUGH ENTERING ITEMS INTO
THE FILE, SIMPLY ENTER 'QUIT' AS YOUR
ONLY COMMAND ON THAT LINE.

WHAT NAME FOR THIS NEW EXEC FILE
(29 CHARACTERS MAXIMUM)
ADVANCED OPERATING SYSTEMS

SLOT (DEFAULT=6): 6

DRIVE (DEFAULT=1): 2

!HOME:PRINT"WE ARE MAKING AN EXEC FILE"



Once you have done all of the above, you will see an exclamation mark (!) as a prompt. You can now enter any commands just as you would at command level, and they will be saved out to the disk. Commas, colons, and quote marks are allowed here, so you can type just about anything you want.

In order to end a creating session, type the word "QUIT" as your command. It must be the only word on the line. You will then be returned to the Main Program Menu.



Chapter 11

EDIT EXEC FILES

Most of the time, it is impossible to get an Exec file correct the first time through. It can be very frustrating and tedious to retype the entire file just to correct one mistake. On the other hand, if you do happen to get it right the first time, it could be considered a miracle and reason for celebration. This is particularly true as your Exec files increase in size.

EDIT EXEC FILES will allow you to load, inspect, change, and rewrite any Exec file on a disk. All you need to do is provide the name of the file and any changes you want to make.

To use this program, all you have to do is choose option 10 from the Main Program Menu. Next you must enter the name of the file to be edited and the Slot and Drive location. The file will then be read into memory and shown on the screen.

```
PRESS CTRL-S TO STOP/START OUTPUT
1HOME:PRINT"WE ARE MAKING AN EXEC FILE"
2CLEAR
3LOAD HELLO
4LIST
5CATALOG
6RUN
7PRINT"THIS IS THEEND OF THE FILE...."
LINE NUMBER TO EDIT (1-8): 7
7PRINT"THIS IS THEEND OF THE FILE...."
TO WHAT: ?"THIS IS THE END OF OUR FILE.
```

This is a line-oriented editor, so if you are changing a line, you must retype it entirely. You can add lines to the end of the file, or you can delete lines from any place within the file. To do this choose the line number you wish to edit. In order to add lines to the end of the file, choose a line number one higher than the last one in the file. For example, if there are 10 lines in your file, enter "11" as the line to edit. You will be notified that you are entering a new line. Then just enter your commands as normal.

In order to end editing, enter "QUIT" as your command, just as in CREATING EXEC FILES. Your edited file will then be saved out to the disk you specified. After this, you will be returned to the Main Program Menu.



Chapter 12

TEXT DUMP

This is a machine-code program that will allow you to dump to printer whatever is on the screen.

To use this program, choose option 11 from the Main Program Menu. Then supply your printer slot. Now, anytime that you press **Ctrl-P**, whatever is on the screen will go to the printer.

This will only work when the Apple is waiting for input from the user. Usually this is anytime you can see the blinking cursor.

This feature will stay in effect until you do one of the following:

1. Press **RESET**
2. Type PR #6
3. Load another machine code program that writes over Page 3 of memory.

Once this routine has been implemented, you will be returned to the Main Program Menu.






Chapter 13

BASE CONVERT

BASE CONVERT works on the premise that it is easier to let the computer do what it was intended to do than to do it yourself. This program works with binary, octal, decimal, and hexadecimal numbers. It will convert a number from any of the above bases to all of the bases.

All you need to do is to enter the base code and then the number in the format "CN". This "C" is the base code, and the "N" is the number to be converted.

For example, if you wish to convert the decimal number 3456, type "D3456" and then press . The computer will respond with:

BASE CONVERT

THIS PROGRAM WILL CONVERT A NUMBER FROM
BINARY, OCTAL, DECIMAL OR HEXADECIMAL
TO BINARY, OCTAL, DECIMAL & HEXADECIMAL.
JUST ENTER B,O,D, OR H AND THEN THE
NUMBER. FOR EXAMPLE, DECIMAL 123 WOULD
BE ENTERED AS 'D123'.

ENTER 'QUIT' AS THE NUMBER TO END THIS
PROGRAM.

(B,O,D,H) NUMBER: D3456

BINARY: 110110000000

OCTAL: 6600

DECIMAL: 3456

HEX: D80

(B,O,D,H) NUMBER: 

Numbers can be converted from any of the other bases simply by using a different base code: "B" for Binary, "O" for Octal, "D" for Decimal, or "H" for Hexadecimal.

In order to exit the program, simply enter the word "QUIT" when prompted for input, and you will be returned to the Main Program Menu.

Chapter 14

EXIT SYSTEM

This is the proper way to exit APPLE-AIDS. It should be noted here that APPLE-AIDS uses error-handling routines that prevent certain things from happening.

Pressing **Ctrl-C** may result in the system hanging indefinitely.

Pressing **RESET** will result in the system rebooting.





Chapter 15

USER NOTES

1. All Advanced Operating Systems programs are extensively tested and debugged. However, you are the final judge. If you feel that you have discovered something that should not be happening (i.e. a "bug"), then WRITE DOWN the conditions and steps that led you to the discovery and apparent problem. Then send us a note and we will figure out what happened.

2. All Advanced Operating Systems programs are covered by our limited warranty. Specific warranty information is contained elsewhere in this manual.

3. The only way that our guarantee will apply is if you fill out and return the user registration card contained in this package. It lets us know that you are indeed the owner of this software package.



GLOSSARY

ASCII: American Standard Code for Information Interchange. A standard 8-bit (1 byte) information code used with most computers and data terminals.

BASIC: Beginners All-purpose Symbolic Instruction Code. A widely used computer language. Particularly popular on micro computers.

BINARY: A numerical system based on the powers of 2. The only digits used in binary are "0" and "1". See also BIT.

BIT: A binary digit. This can either be a "0" or a "1".

BOOT: Short for "Boot-strap". Refers to the process of bringing DOS and a disk system on line. On the Apple, this includes reading DOS from tracks 0, 1, and 2 and executing the "Hello" program (if any).

BYTE: Eight bits. A byte may have any value of 0 through 255.

CATALOG: DOS command that prints the directory of files to the current output device. Also interchangeable with DIRECTORY when referring to a physical section of a disk.

CONTROL CHARACTERS: Any ASCII character in the value range of \$01 through \$1F. Most of these characters can be typed from the Apple keyboard, but they will not appear on the CRT. They are used to control certain functions such as line feed, carriage return, and bell output.

CPU: Central Processing Unit. The primary processor for a computer. It contains the arithmetic units, logic units, and the status registers for the computer. In the Apple II, the CPU is the 6502.

CRT: Cathode Ray Tube. A display monitor frequently used with computers. A television set is a type of CRT.

DECIMAL: A numerical system based on the Powers of 10. Uses the digits "0" through "9".

DIRECTORY: Normally contained on track 17 (\$11), sectors 1-15 (\$01-\$0F) of a diskette. Contains information that allows DOS to locate and check the status of individual files.



DISASSEMBLE: A term used to describe the process of translating machine code to assembly language.

DOS: Disk Operating System. The set of commands and instructions that permit the orderly flow of data and programs between a mass storage peripheral (a disk subsystem) and the CPU.

EPROM: Erasable Programmable Read Only Memory.

EXEC FILE: A text file containing commands or program lines to be executed by the Apple. The commands stored appear just as they would if they were typed at command level from the keyboard.

FIRMWARE: Those programs and instructions that reside in ROM or PROM for use on a computer system.

HARDWARE: The physical circuits and materials that make up a computer or its peripherals.

HEXADECIMAL: A numerical system based on the powers of 16. Hexadecimal is frequently used in computers as an alternative to BINARY because of its relative ease of understanding. The digits "0" through "9" and the letters "A" through "F" are used. For example, "2E" means 14 ones and 2 sixteens, or 46 (decimal).

K: Kilobyte. A unit of measure referring to 1024 Bytes.

MONITOR: See CRT.

MONITOR: The series of instructions written in machine code that take first control of the computer when it is powered up. The monitor on the Apple resides in ROM.

PERIPHERAL: A device that can be attached to a computer to enhance its inherent abilities. Some examples are printers and disk drives. Most peripherals are considered Input/Output devices.

POWER-UP: The process of applying power to a computer to start it up. In plain English, this involves flipping the switch to supply power to the computer.

PROM: Programmable Read Only Memory.

RAM: Random Access Memory. The main storage and work area of a computer. RAM is usually measured in K.



ROM: Read Only Memory.

SECTOR: A storage area used by DOS to store data on a disk. Each Track on a disk is divided into 16 (DOS 3.3) or 13 (DOS 3.2) sectors. Each sector can store 256 bytes of data.

SOFTWARE: Computer programs that can be loaded into RAM and contain a series of executable steps for the computer to process.

TOGGLE: A "switch" used to turn something on and off. In computers, this can be a command or a memory location. For example, accessing location \$C030 will produce an audible click on the speaker of the Apple.

TRACK: A sequence of binary cells arranged on a disk so as to enable data to be stored and retrieved in a logical manner. There are 35 tracks in standard Apple DOS. See also SECTOR.

TRACK BIT MAP: The section of VTOC that tells DOS which tracks and sectors are free.

TRACK/SECTOR LIST: A sector on the disk which points to the individual tracks and sectors occupied by any given file on that disk. Each file has its own track/sector list.

UTILITY: A program developed for the express purpose of aiding the development or diagnosis of other programs, data files, or disk files.

VTOC: Volume Table Of Contents. Track 17 (\$11), sector 0 on a normally formatted Apple diskette. The VTOC contains information that tells DOS which tracks and sectors are in use and where the first directory sector is located.



Appendix A

VTOC

(Volume Table of Contents)

The VTOC, located at Track \$11, Sector \$0, is used by DOS to locate the first directory sector and to tell which tracks and sectors on the disk are free for use. It also contains certain other controlling information that is essential to the proper operation of DOS.

VTOC FORMAT

BYTE	PURPOSE/CONTENT
\$00	Not used
\$01	Track number of first directory track (usually \$11)
\$02	Sector number of first directory sector (usually \$0F)
\$03	Release number of DOS
\$04-\$05	Not used
\$06	Diskette volume number (\$01-\$FE)
\$07-\$26	Not used
\$27	Maximum track/sector pairs per Track/Sector List sector (normally \$7A)
\$28-\$2F	Not used
\$30	Last track where sectors were used
\$31	Track allocation direction (+1 or -1)
\$32-\$33	Not used
\$34	Tracks per diskette
\$35	Sectors per track
\$36-\$37	Bytes per sector (L-H)
\$38-\$3B	Track 0 bit map
\$3C-\$3F	Track 1 bit map
\$40-\$43	Track 2 bit map
\$44-\$47	Track 3 bit map
\$48-\$4B	Track 4 bit map
\$4C-\$4F	Track 5 bit map
\$50-\$BB	Bit maps for tracks 6 through \$20
\$BC-\$BF	Track \$21 bit map
\$C0-\$C3	Track \$22 bit map
\$C4-\$FF	Not presently used

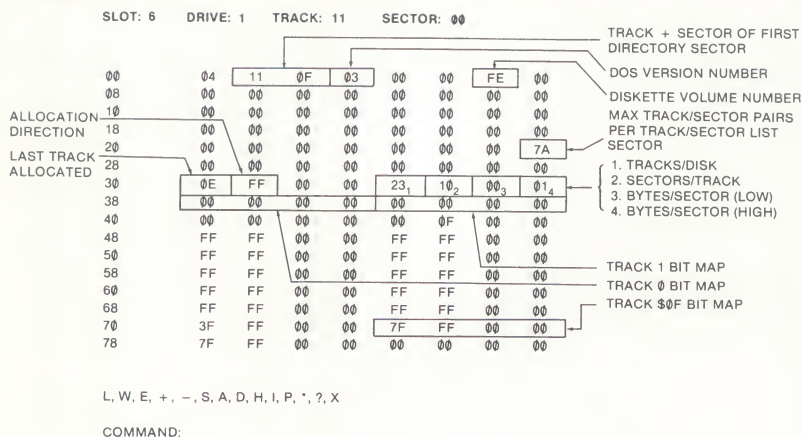


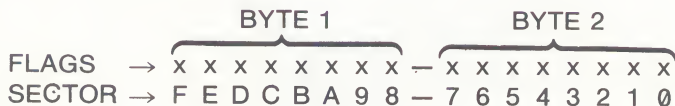
Fig. 2—First part of VTOC Sector

NOTES

1. In theory the diskette volume number in byte \$06 can only be of value \$01 through \$FE. In practice, with special programs, you could use \$0 or \$FF, but using volume \$00 will not allow linking and running of a "Hello" program. In other words, booting up a disk with a volume number of "0" will not run a "Hello" program. You will be returned to BASIC when DOS has been loaded and the DOS hooks established.

2. The volume number is also stored on each track and sector header. Changing this byte alone will NOT change the volume of a diskette.

3. Track bit maps are allocated 4-bytes per track, a total of 140 bytes. The second 2 bytes are not used. The first 2 bytes are used in the following manner:



A "0" in any position denotes that the sector is in use; a "1" denotes that the sector is free.



Appendix B

THE DISKETTE DIRECTORY

Every normally formatted Apple diskette has a directory. This is essential so that DOS can find any file that you or a program asks for. The directory is laid out in a very systematic manner, and can be viewed and adjusted by a user with a little practice.

The directory starts at Track \$11, Sector \$0F. Each directory sector can contain 7 entries. This first directory sector is linked to the second directory sector at location \$11-\$0E), which is linked to the sector below it, and so on. The last directory sector is located at (\$11-\$01). This is a total of 15 sectors. At 7 entries per sector, there is a 105 file directory entry maximum on a normal diskette.

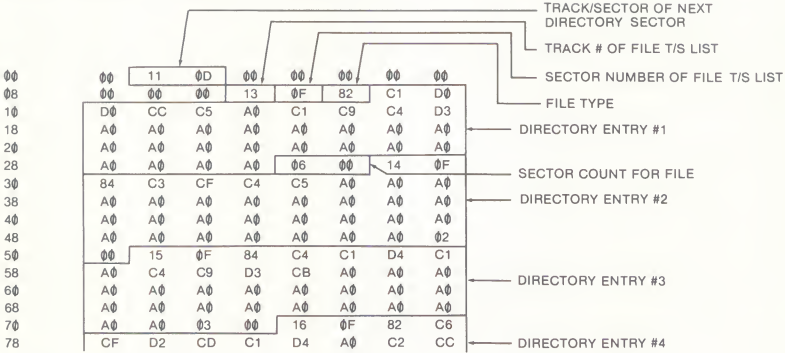
Each directory sector is configured in the same manner, so we will describe just one sector here.

DIRECTORY SECTOR FORMAT

BYTE	PURPOSE/CONTENTS
\$00	Not Used
\$01	Track for next catalog sector
\$02	Sector for next catalog sector
\$03-\$0A	Not Used
\$0B-\$2D	First directory entry
\$2E-\$50	Second directory entry
\$51-\$73	Third directory entry
\$74-\$96	Fourth directory entry
\$97-\$B9	Fifth directory entry
\$BA-\$DC	Sixth directory entry
\$DD-\$FF	Seventh directory entry

NOTE: Bytes \$01 and \$02 will be set to 0, if the particular sector is the last directory sector.

SLOT: 6 DRIVE: 1 TRACK: 11 SECTOR: 0E



L, W, E, +, -, S, A, D, H, I, P, *, ?, X

COMMAND:

a) Hex.

SLOT: 6 DRIVE: 1 TRACK: 17 SECTOR: 14

0	@	Q	M	@	@	@	@
8	@	@	@	S	O	B	A P
16	P	L	E		A	I	D S
24							
32							
40					F	@	T O
48		D	C	O D	E		
56							
64							
72							B
80	@	U	O	D	D	A	T A
88		D	I	S	K		
96							
104							
112			C	@	V	O	B F
120	O	R	M	A	T	B	L

L, W, E, +, -, S, A, D, H, I, P, *, ?, X

COMMAND:

b) ASCII.

Fig. 3—Sample Directory Sector

Each directory entry above is laid out in the following format.



DIRECTORY ENTRY FORMAT

BYTE	PURPOSE/CONTENTS
\$00	Track of first track/sector list for the file
\$01	Sector of first track/sector list for the file
\$02	File type and lock/unlock notation (see below)
\$03-\$20	File name (30 characters)
\$21-\$22	Number of sectors used by file

NOTES

1. All of the byte references listed previously are RELATIVE. This means that if you are referring to the fourth directory entry, you would add the above byte locations to the entry offset. The fourth directory entry begins at byte \$74 of the directory sector. The file type would be noted at byte \$76 (\$74 + \$02).

2. If a file is deleted, bytes \$00 and \$20 are affected. See the section on deleting files.

3. Byte \$02 may have any of the following values:

LOCKED	—	UNLOCKED	—	FILE TYPE
\$00		\$80		Text
\$01		\$81		Integer
\$02		\$82		Applesoft
\$03		\$84		Binary
\$08		\$88		S type file
\$10		\$90		Relocatable object file
\$20		\$A0		A type file
\$40		\$B0		B type file

4. Even though 2 bytes are used to notate the number of sectors used by a file, only the low-order byte (\$21) is used by the CATALOG function to show the number of sectors used by the file.



Appendix C

TRACK SECTOR LISTS**(Or, Where Do We Go From Here?)**

Each file on a disk has a Track Sector List. This is a special sector on the disk that allows DOS to know which sectors come next for an individual file. In different computer systems, there are different methods of linking file sectors together. Some use pointers from one sector to the following sector, much the same way as Apple DOS links directory sectors. Others keep special lists for each file. That is what a track sector list contains. It is just a lot of numbers that point to where a file is physically located on a disk.

The following is the format for a Track Sector List sector.

TRACK/SECTOR LIST FORMAT

BYTE	CONTENTS/PURPOSE
\$00	Not used
\$01	Track number pointer to next T/S List sector ("0" if no link)
\$02	Sector number pointer to next T/S List sector ("0", if no link)
\$03-\$04	Not used
\$05-\$06	Sector offset for first sector in list
\$07-\$0B	Not used
\$0C-\$0D	Track and sector of next file sector
\$0E-\$0F	Track and sector of next file sector
\$10-\$FF	Track and sector pairs (up to 120 more)

NOTE: The sector offset in bytes \$05-\$06 is used as a check to make sure that the entire file is being read correctly. If another Track/Sector List sector is needed, the link would be noted in bytes \$01-\$02. Since the first T/S pair noted in this second sector would actually be the 123rd sector of the file, this information would be stored in the offset bytes.



SLOT: 6 DRIVE: 1 TRACK: 1C SECTOR: 0F

00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

L, W, E, +, -, S, A, D, H, I, P, *, ?, X

COMMAND:

Fig. 4—Sample Track/Sector List Sector.



Appendix D

HOW DOS DELETES A FILE**(And How We Can Undo It)**

When a user or a program issues a DELETE command, the disk file is not completely erased from the disk. In a certain sense, it is never fully taken off of the disk unless it is written over or the disk is initialized again.

First of all, DOS, upon receipt of the DELETE command, will mark the directory entry for the file by moving relative byte 0 to relative byte 32 (\$20) and setting relative byte 0 to the value 255 (\$FF). This is shown in Fig. 3 on page 50. Now the file will not appear when you ask for a "CATALOG" of the disk.

But this is not all, for now DOS updates the track bit map (VTOC) for any tracks on which the file may have been saved. Now, for all practical purposes, the file is gone, because next time you save something, it will be written into the space previously occupied by the file you just deleted.

The actual physical sectors have not been erased unless you write over them or initialize the disk. If you save another file to the disk, then all of the pointers to the old file location will be removed or erased, at this point.

As long as the file pointers have not been removed, the file (usually) is still recoverable. Using a program like UNDELETE FILES accomplishes this goal. We said usually, because, in theory, all that is needed is to restore relative byte \$0 to its "pre-deletion" value (remember, this value is stored in relative byte \$20). Then the track bit maps need to be adjusted to show the file sectors in use again. Once this is done, the file is effectively restored.

The problems arise if disk usage has occurred since the file was first deleted. When you increase the size of an existing file, or save a new file to disk, chances are that it will be written into part, if not all, of the space previously occupied by the deleted file. If you are



saving a new file, its directory entry will be written to the space that used to be occupied by the file you deleted.

Undeleting files is tricky, but manageable if the above considerations, as well as those pointed out in Chapter 7, are kept in mind.

Appendix E

DISK POTPOURRI

There are several interesting locations on a disk that can be modified to achieve results that the user may desire.

This is NOT a comprehensive list. With a little bit of time and a minimum amount of trouble, you can probably find many more yourself.

TRACK	SECTOR	BYTE	USE
1	7	\$84	Beginning of DOS command table (text)
1	8	\$74	Beginning of DOS error message table (text)
1	9	\$75	Name of user's "Hello" program stored here
1	9	\$B8	Name of program to be run, if "Hello" program is an Applesoft® program and that language is not resident in memory.
1	\$C	\$A4	Number of lines to print before waiting for key press when printing disk directory (when user gives "Catalog" command)
2	2	\$AF	Text "DISK VOLUME" stored (backward). Used when "Catalog" command is issued.
\$11	0	ALL	VTOC
\$11	1	ALL	Last directory sector
\$11	\$F	ALL	First directory sector

Appendix F

APPLE ASCII CHART

Apple computer, as we all know by now, does not use lower-case letters, so the normal ASCII code was modified slightly to allow certain special functions on the Apple display. These functions are the FLASH and INVERSE commands of BASIC. Also, certain disk text storage of commands and information requires that the high bit be "on" in order that the information is translated correctly and used properly. For example, in the DOS command table (text), the DOS commands are in normal ASCII, except for the last character of each command, which has the high bit set. This way, DOS knows when one command ends and the next one starts.

A chart follows showing ASCII codes with numeric values and keyboard values. The numeric values are given in decimal, hex, enhanced decimal, and enhanced hex. In this instance, "enhanced" signifies that the high bit is on in the value given.

These are ASCII values for the Apple, so the lower case character values have been left off.

Appendix G

ASCII CODE TABLE

DEC	HEX	ENH DEC	ENH HEX	CHAR	KEYBOARD
0	00	128	80	NULL	ctrl-@
1	01	129	81	SOH	ctrl-A
2	02	130	82	STX	ctrl-B
3	03	131	83	ETX	ctrl-C
4	04	132	84	ET	ctrl-D
5	05	133	85	ENQ	ctrl-E
6	06	134	86	ACK	ctrl-F
7	07	135	87	BEL	ctrl-G
8	08	136	88	BS	ctrl-H
9	09	137	89	HT	ctrl-I
10	0A	138	8A	LF	ctrl-J
11	0B	139	8B	VT	ctrl-K
12	0C	140	8C	FF	ctrl-L
13	0D	141	8D	CR	ctrl-M
14	0E	142	8E	SO	ctrl-N
15	0F	143	8F	SI	ctrl-O
16	10	144	90	DLE	ctrl-P
17	11	145	91	DC1	ctrl-Q
18	12	146	92	DC2	ctrl-R
19	13	147	93	DC3	ctrl-S
20	14	148	94	DC4	ctrl-T
21	15	149	95	NAK	ctrl-U
22	16	150	96	SYN	ctrl-V
23	17	151	97	ETB	ctrl-W
24	18	152	98	CAN	ctrl-X
25	19	153	99	EM	ctrl-Y
26	1A	154	9A	SUB	ctrl-Z
27	1B	155	9B	ESCAPE	ESC
28	1C	156	9C	FS	n/a
29	1D	157	9D	GS	ctrl-shift-M
30	1E	158	9E	RS	ctrl-^
31	1F	159	9F	US	n/a
32	20	160	A0	SPACE	SPACE



DEC	HEX	ENH DEC	ENH HEX	CHAR	KEYBOARD
33	21	161	A1	!	!
34	22	162	A2	"	"
35	23	163	A3	#	#
36	24	164	A4	\$	\$
37	25	165	A5	%	%
38	26	166	A6	&	&
39	27	167	A7	'	'
40	28	168	A8	((
41	29	169	A9))
42	2A	170	AA	*	*
43	2B	171	AB	+	+
44	2C	172	AC	,	,
45	2D	173	AD	-	-
46	2E	174	AE	.	.
47	2F	175	AF	/	/
48	30	176	B0	0	0
49	31	177	B1	1	1
50	32	178	B2	2	2
51	33	179	B3	3	3
52	34	180	B4	4	4
53	35	181	B5	5	5
54	36	182	B6	6	6
55	37	183	B7	7	7
56	38	184	B8	8	8
57	39	185	B9	9	9
58	3A	186	BA	:	:
59	3B	187	BB	;	;
60	3C	188	BC	<	<
61	3D	189	BD	=	=
62	3E	190	BE	>	>
63	3F	191	BF	?	?
64	40	192	C0	@	@
65	41	193	C1	A	A
66	42	194	C2	B	B
67	43	195	C3	C	C
68	44	196	C4	D	D
69	45	197	C5	E	E

ASCII CODE TABLE CONT.

DEC	HEX	ENH DEC	ENH HEX	CHAR	KEYBOARD
70	46	198	C6	F	F
71	47	199	C7	G	G
72	48	200	C8	H	H
73	49	201	C9	I	I
74	4A	202	CA	J	J
75	4B	203	CB	K	K
76	4C	204	CC	L	L
77	4D	205	CD	M	M
78	4E	206	CE	N	N
79	4F	207	CF	O	O
80	50	208	D0	P	P
81	51	209	D1	Q	Q
82	52	210	D2	R	R
83	53	211	D3	S	S
84	54	212	D4	T	T
85	55	213	D5	U	U
86	56	214	D6	V	V
87	57	215	D7	W	W
88	58	216	D8	X	X
89	59	217	D9	Y	Y
90	5A	218	DA	Z	Z
91	5B	219	DB	[n/a
92	5C	220	DC	\	n/a
93	5D	221	DD]	shift-M
94	5E	222	DE	^	^
95	5F	223	DF	—	n/a



INDEX

A

Applesoft® , 4, 15
ASCII, 22-23, 26, 44, 57-60

B

Back-up copies, 3, 7
Base convert, 39-40
BASIC, 11, 44
Binary, 15-16, 39, 44
Bootling, 3-4, 9, 11, 44, 48
Byte, 22, 28, 44

C

Catalog, 9, 16, 44, 56
Create exec files, 33-34
CRT, 3, 23, 26, 44
Ctrl-C, 41
Ctrl-P, 37
Ctrl-S, 27

D

Decimal, 17, 22-24, 39, 44, 57
Delete, 16, 19, 51, 54-55
Directory, 19, 44, 47, 49-51
Disk
 copy, 31
 editor, 21-29
 command descriptions, 21-29
 command line, 21
 disassembler, 27
 display modes, 23-27
 display page, 23, 25
 DOS 3.3/3.2 Editor differences, 20
 edit types, 22
 input, 28
 status, 28-29
 map, 13-14
DOS versions, 7

E

Edit exec files, 35-36
Exec files, 33-36, 45

F

Files
 code, 16
 length, 16
 sectors used, 17
 text, 16
 types, 16, 51
 undeleted, 19
Flashing, 23, 57
Format blank disk, 9, 11

G

Getting started, 3-4

H

Hexadecimal, 22, 24-25, 40, 45, 57

I

Improved directory, 16
Init, 9
Inverse, 17, 23, 52

L

Language card, 4

M

Main program menu, 3
Map, disk, 13
Master Create, 11



Mnemonics, 26
Monitor, 19, 45

O

Octal, 40

P

Pirates, 31
Printer, 27, 37
Program menu, 3
Purpose of copy programs, 31

R

Reset, 37, 40

Q

Quit, 34, 36, 40

S

Safety precautions, 7
Sector listings, 17
Sectors
 definition, 46

free, 14
in-use, 13
pointer, 22-23

T

Text dump, 37-38
Toggle, 23, 27, 46
Track
 0, 9
 1, 9, 11
 2, 9, 11
 17, 9
 allocation order, 48
 bit map, 19, 48, 54
 borders, 23
 copy, 31
 definition, 46
 /sector list, 46, 51

U

Undeleted files, 19
User notes, 43

V

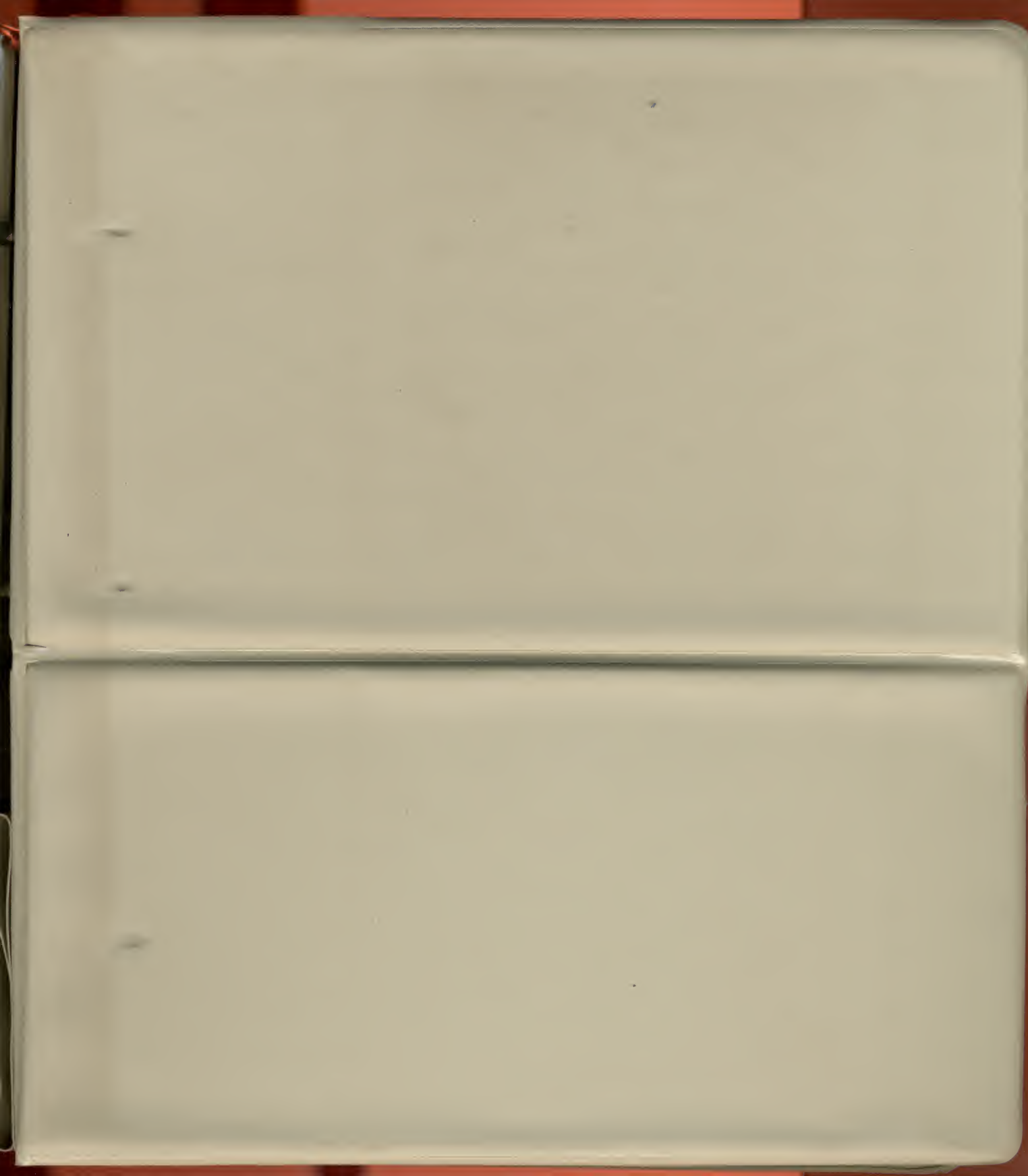
VTOC, 9, 46-48, 56

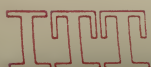


Allen Wyatt has been actively involved with the microcomputer industry for four years and is currently a program consultant and editor with Advanced Operating Systems in Michigan City, Indiana. Mr. Wyatt also acts as a computer consultant to several private businesses in the northern Indiana area.

He has written several commercial software packages for the Apple computer. The broad range of programs runs the gamut from small system data bases to games and utilities. APPLE-AIDS is a fine example of the latter category.

Besides being a computer author, Al is a devoted family man and active church member. He uses his personal Apple computer to assist him in all of these areas. At home, his family spends many hours using the computer every day.





HOWARD W. SAMS & CO., INC.

Developed by Advanced Operating Systems®

4300 W. 62nd Street

Indianapolis, Indiana 46268

(317)298-5566